UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/882,525	06/15/2001	Paul Egli	LS/0016.00	9946

8791 7590 08/25/2006

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
RAMPURIA, SATISH	

ART UNIT	PAPER NUMBER
2191	

DATE MAILED: 08/25/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/882,525	EGLI, PAUL	
	Examiner	Art Unit	
	Satish S. Rampuria	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 June 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-45* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-45* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Response to Amendment

1. This action is in response to the RCE received on June 22, 2006.

2. Claims amended by the Applicant: 1 and 41.

3. Claims pending in the application: 1-45.

Response to Arguments

4. Applicant's arguments with respect to claims have been considered but they are not persuasive.

In the remarks, the applicant has argued that:

(i) The Applicant respectfully submits that Rollins and Claussen, alone or in combination, fails to teach or suggest "creating an object oriented programming (OOPL) class that extends the abstract command tag... thereby creating a corresponding customized command tag [and] invoking the customized command tag". Claussen merely describes handling custom tags but fails to describe extending an abstract command tag thereby creating a corresponding customized command tag. Although Rollins describes generating Java classes which mediate communication between a user and an XML document, the classes are not described as being classes for tags. Further, at no point does Rollins refer to, or suggest, the use and extension of an abstract command tag. (Remarks, page 12-13).

(ii) The Applicants respectfully disagree. Rollins discloses java classes which are isolated from a provided library of user interface classes, so that the subgroup of classes is capable of performing a customized rule in the Rollins interface (Rollins, page 3,

paragraph [0038]; page 2, paragraph [0015]). Thus Rollins merely describes a subgroup of predefined classes. The abstract command tag, however, is a java element that is extended with new code, but not a regrouping of existing code, to provide for custom functionality of the abstract tag. Because extending a class is not the same as creating a subgroup of existing classes, Rollins fails to describe or suggest an abstract command tag or extending the abstract command tag. (Remarks, page 13).

(iii) Neither Rollins nor Claussen, alone or in combination, teaches or suggest "providing a Web application development framework, said framework including an abstract command tag that predefines at least some generic Web application activities [and] an object oriented programming language (OOPL) class that extends the abstract command tag for providing execution logic for said at least one custom action," as claimed in Claim 1. Thus, claims 2-12, 15, and 17-20, which depend from claim 1, are also not rendered obvious by Rollins in view of Claussen. Therefore, the Applicants respectfully request withdrawal of the rejection of claims 1-12, 15, and 17-20 under 35 U.S.C. 103. (Remarks, page 14).

(iv) As discussed above with respect to claim 1, neither Rollins nor Claussen, alone or in combination teach or suggest the use and extension of an abstract command tag. Rollins merely discusses creating Java classes for facilitating communication between a user and an XML document, while Claussen simply handles custom tags. Therefore, since neither reference, alone or in combination teaches or suggests the use and extension of an abstract command tag, claim 21 is not rendered obvious by Rollins in

view of Claussen. Furthermore, claims 22-32, 35, 37-40 depend on claim 21, and include additional features and limitations. Thus, claims 22-32, 35, 37-40 are also not obvious under Rollins in view of Claussen. (Remarks, page 16).

Examiner's response:

(i) In response to the Applicant's argument, both Rollins and Claussen disclose method and system to be used by Web applications and/or develop Web applications. More specifically, Rollins discloses XML application development, which is done on the Web environment and used by the World Wide Web (page 2 [0015-0016]). Claussen discloses processing the custom tag in a document object mode (DOM) representation that is an internal XML document data structure representation and basically a tree of all nodes in an XML file (col. 3, lines 14-52). A set of Java classes disclosed by Rollins is well known in the art that Java classes are Object Oriented Programming Language (OOPL). Applicants note that Rollins discloses Java classes that are capable of performing a customized rule, Remarks page 13. Further, Claussen discloses a custom tag and tag handler to execute the custom tags. Thus, it is submitted that the combination of Rollins and Claussen does disclose the limitations "creating an object oriented programming (OOPL) class that extends the abstract command tag... thereby creating a corresponding customized command tag [and] invoking the customized command tag". Applicant only makes general allegations and does not point out any errors in the rejection. Rather, in response to applicant's arguments against the references individually, one cannot show nonobviousness by

attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Therefore, the rejection is proper and maintained herein.

(ii) In response to the Applicants argument, that Rollins fails to describe or suggest an abstract command tag or extending the abstract command tag and further explains that the abstract command tag, however, is a java element that is extended with new code, but not a regrouping of existing code, to provide for custom functionality of the abstract tag. However, specification does not support tah a java element that is extended with new code, rather it supports on paragraph [0021] of the publication that "Abstract classes: An abstract class is an object-oriented programming class that is not completely implemented and defined, and, therefore, is but an abstract manifestation of a class that cannot compile into an instance of a runtime object. Developers must extend, and fully define, a subclass derived from an abstract class to use it." Which is defining a class in a manner that compile into an instance of a runtime object, where Rollins has a set of Java classes designed to and ready for compile. Therefore, the rejection is proper and maintained herein.

(iii) The response to the argument (ii) applies here as well.

(iv) The response to the argument (i) and (ii) applies here as well.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claim 1-12, 15, 17-20, 21-32, 35, 37-41, 43, and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Publication No. 2002/0129060 to Rollins et al. (hereinafter called Rollins) in view of US Patent No. 6,675,354 to Claussen et al. (hereinafter called Claussen).

Per claims 1 and 41:

Rollins disclose:

- providing a Web application development framework (see the title), said framework including an abstract command tag that predefines at least some generic Web application activities (page 2, paragraph 15 "based upon an XML schema and a set of user customization rules");
- specifying at least one custom action (page 2, paragraph 15 "a set of user customization rules") that is desired to be performed by a Web application (page 2; paragraph 15 "produce a set of components that interact to provide a user-specific... XML document");
- creating an object-oriented programming language (OOPL) class that extends the abstract command tag for providing execution logic for said at least one custom action (page 3, paragraph 38 "a set of Java classes designed to mediate communication between the user and the synchronized tree manager"), in addition to pre-existing logic that supports said at least some generic Web application activities, thereby creating a

corresponding customized command tag that is capable of being embedded within a Web page (page 3, paragraph 38 "a set of Java classes designed to mediate communication between the user and the synchronized tree manager")

- embedding the customized command tag in a Web page of the Web application (page 2, paragraph 34 "XML data ... allows access for all users despite input/output restrictions").

Rollins does not e xplicitly disclose upon execution of the Web application including an embedded customized command tag in a Web page, invoking the customized command tag for conditionally executing said specified at least one custom action based on run-time conditions.

However, Claussen discloses in an analogous computer system executing the Web application, including invoking the customized command tag for conditionally executing said specified at least one custom action based on run-time conditions (col. 3, lines 31-42 "Upon encountering a custom tag, an appropriate tag handler... is invoked... a tag registration routine is used for recognizing... if the name does not match one of the registered tags, the routing converts the name... If the tag recognition routine recognizes the name... it converts the attributes to the appropriate case... hands the resulting element off to a correct handler for processing").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method to execute the appropriate tag as taught by Claussen into the method of developing web applications as taught by Rollins. The modification would be obvious because of one of ordinary skill in the art would be motivated to implement only those tags which are needed to provide new techniques for publishing Internet content that

can fully leverage the manipulation and template mechanism of XSLT with the scripting capability of the JSP/SAP model as suggested by Claussen (col. 3, lines 7-11).

Per claim 2:

- wherein said run-time conditions include run-time parameters specified during invocation of the customized command tag. The limitations in the claims are similar to those in claim 1, and rejected under the same rational set forth in connection with the rejection of claim 1.

Per claim 3:

The rejection of claim 2 is incorporated, and further, Rollins disclose:

- wherein said run-time parameters are specified via Hypertext Transport Protocol (HTTP) parameters, during invocation of the customized command tag (page 1, paragraph 10 "XML... deliver this data by use of the standard HTTP protocol... layer protocol").

Per claim 4:

The rejection of claim 1 is incorporated, and further, Rollins disclose:

- wherein said abstract command tag comprises an abstract base class (page 3, paragraph 38 "user... specify a set of customization rules... the result of code-generation is a set of Java classes...").

Per claim 5:

- wherein said abstract command tag includes an abstract execute method. The limitations in the claims are similar to those in claim 4, and rejected under the same rational set forth in connection with the rejection of claim 4.

Per claim 6:

- wherein said abstract execute method is overridden during creation of the customized command tag, for defining a customized execute method providing specific runtime execution logic for the customized command tag. The limitations in the claims are similar to those in claim 4, and rejected under the same rational set forth in connection with the rejection of claim 4.

Per claim 7:

- wherein creation of the OOPL class that extends the base class includes providing an implementation for the abstract execute method. The limitations in the claims are similar to those in claim 4, and rejected under the same rational set forth in connection with the rejection of claim 4.

Per claim 8:

The rejection of claim 1 is incorporated, and further, Rollins does not explicitly disclose wherein said customized command tag includes an ability to conditionally affect application flow based on results obtained from a specified action.

However, Claussen discloses in an analogous computer system wherein said customized command tag includes an ability to conditionally affect application flow based on results obtained from a specified action (col. 3, lines 31-42 "Upon encountering a custom tag, an appropriate tag handler... is invoked... a tag registration routine is used for recognizing... if the name does not match one of the registered tags, the routing converts the name... If the tag recognition routine recognizes the name... it converts the attributes to the appropriate case... hands the resulting element off to a correct handler for processing").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method to execute the appropriate tag as taught by Claussen into the method of developing web applications as taught by Rollins. The modification would be obvious because of one of ordinary skill in the art would be motivated to implement only those tags which are needed to provide new techniques for publishing Internet content that can fully leverage the manipulation and template mechanism of XSLT with the scripting capability of the JSP/SAP model as suggested by Claussen (col. 3, lines 7-11).

Per claim 9:

- wherein application flow is affected by routing to a particular Web page. The limitations in the claims are similar to those in claim 8, and rejected under the same rational set forth in connection with the rejection of claim 8.

Per claim 10:

- wherein said result obtained is either success or failure. The limitations in the claims are similar to those in claim 8, and rejected under the same rational set forth in connection with the rejection of claim 8.

Per claim 11:

- wherein application flow is directed to a first page if a success is obtained as the result, and is directed to a second page if a failure is obtained as the result. The limitations in the claims are similar to those in claim 8, and rejected under the same rational set forth in connection with the rejection of claim 8.

Per claims 12 and 15:

The rejection of claim 8 is incorporated, and further, Rollins disclose:

- wherein said application flow includes routing to a different page than is currently displayed in a user's browser (page 3, paragraph 36 "generating multiple customizable interfaces for XML documents").

Per claims 17 and 18:

The rejection of claim 1 is incorporated, and further, Rollins disclose:

- wherein said customized command tag is invoked when an end user activates a link that points to a Web page containing the customized command tag (page 3, paragraph 48 "The Renderer defines the concept of a cursor... of the registered mediators should be rendering the portion of the tree pointed to by the cursor. When the cursor is moved, the

new view of the tree should be rendered... a mediator will have to move the cursor more than one time to achieve the desired view...").

Per claim 19:

The rejection of claim 1 is incorporated, and further, Rollins does not explicitly disclose wherein said Web page containing the customized command tag comprises a JSP (JavaServer Page) compatible page.

However, Claussen discloses in an analogous computer system wherein said Web page containing the customized command tag comprises a Web page generated using dynamic scripting capability (col. 6, lines 18-20 "custom tags are registered through an XML... according to JSP 1.0 specification").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of using JSP compatible page as taught by Claussen into the method of developing web applications as taught by Rollins. The modification would be obvious because of one of ordinary skill in the art would be motivated to implement only those tags which are needed to provide new techniques for publishing Internet content that can fully leverage the manipulation and template mechanism of XSLT with the scripting capability of the JSP/SAP model as suggested by Claussen (col. 3, lines 7-11).

Per claim 20:

The rejection of claim 1 is incorporated, and further, Rollins does not explicitly disclose compiling the Web page generated using dynamic scripting capability into a servlet, said servlet corresponding to said created OOPL class that extends the abstract command tag.

However, Claussen discloses in an analogous computer system compiling the JSP-compatible page into a servlet, said servlet corresponding to said created Java class that extends the abstract command tag (Fig. 2 and col. 6, lines 14-18 "routine continues... to gather all jsp:directives.page tags to ensure a consistent state.. jsp tag libraries (which provide support for JSP 1.0 mechanism)").

The feature of compiling the JSP-compatible page into a servlet would be obvious for the reasons set forth in the rejection of claim 1.

Claims 21-32, 35, and 37-40 are the system claims corresponding to method claims 1-12, 15, and 17-20 respectively, and rejected under the same rational set forth in connection with the rejection of claims 1-12, 15, and 17-20 respectively, above.

Per claim 43:

- wherein said set of OOPL classes run in a Java Virtual Machine (JVM™), wherein the JVM™ is an interpreter that interprets OOPL bytecodes into machine code. The limitations in the claims are similar to those in claim 19, and rejected under the same rational set forth in connection with the rejection of claim 19.

Per claim 44:

- wherein said JVM™ is running at a Web server site. The limitations in the claims are similar to those in claim 19, and rejected under the same rational set forth in connection with the rejection of claim 19.

7. Claims 13, 14, 16, 33, 34, 36, 42, and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rollins and Claussen in view of US Patent No. 6,760,748 to Hakim (hereinafter called Hakim).

Per claims 13 and 42:

The rejection of claim 1 is incorporated, and further, neither Rollins nor Claussen disclose wherein said generic Web application activities include error recording.

However, Hakim discloses in an analogous computer system wherein said generic Web application activities include error recording (col. 44, lines 38-39 "station sample link conditions if 'Roaming' is enabled, transmission errors are recorded").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of recording errors for the network activities as taught by Hakim into the method of developing web application as taught by the combination system by Rollins and Claussen. The modification would be obvious because of one of ordinary skill in the art would be motivated to record the errors to provide the appropriate feedback for different types to of questions as suggested by Hakim (col. 2 and 3, lines 58-67 and 1-14).

Per claims 14, 16, and 45:

The rejection of claim 1 is incorporated, and further, neither Rollins nor Claussen disclose wherein said generic Web application activities include filtering of requests.

However, Hakim discloses in an analogous computer system wherein said generic Web application activities include filtering of requests (col. 29, lines 40-43 "With the addition of optional components (plug-ins), it is possible to extend their functionality to perform detailed content filtering, report generation").

The feature of filtering the requests would be obvious for the reasons set forth in the rejection of claim 13.

Claims 33, 34, and 36 are the system claims corresponding to method claims 13, 14, and 16 respectively, and rejected under the same rational set forth in connection with the rejection of claims 13, 14, and 16 respectively, above.

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Satish S. Rampuria** whose telephone number is **(571) 272-3732**. The examiner can normally be reached on **8:30 am to 5:00 pm** Monday to Friday except every other Friday and federal holidays. Any inquiry of a general nature or relating to the status of this application should be directed to the **TC 2100 Group receptionist: 571-272-2100.**

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, **Wei Y. Zhen** can be reached on **(571) 272-3708**. The fax phone number for the organization where this application or proceeding is assigned is **571-273-8300**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Satish S. Rampuria
Patent Examiner/Software Engineer
Art Unit 2191

WEI ZHEN
SUPERVISORY PATENT EXAMINER